

Jouko Vartiainen

# Sisu-moottorin vikakoodien indikointi ja kääntäminen

Insinööri (AMK)

Tietotekniikka

Kevät 2015



KAJAANIN  
AMMATTIKORKEAKOULU  
UNIVERSITY OF APPLIED SCIENCES

## TIIVISTELMÄ

**Tekijä:** Vartiainen Jouko

**Työn nimi:** Sisu-moottorin vikakoodien indikointi ja kääntäminen

**Tutkintonimike:** Insinööri (AMK), tietotekniikka

**Asiasanat:** CAN-väylä, metsäkoneautomaatio, Omron

Tämän insinöörityön tavoitteena oli välittää harvesterin moottorilta tulevat vikakoodit kuljettajan tietoisuuteen. Työn tilannut ProfiPro Oy halusi päivittää CAN-väylän kautta luettavat vikakoodit uusimman moottorinvalmistaja Agco Powerin EEM4-spesifikaation (moottorin elektroninen hallinta) mukaisiksi. Spesifikaatio pohjautuu raskaissa ajoneuvoissa käytettävään CAN-väylän SAE J1939-standardiin. Tämän lisäksi vikakoodit haluttiin kääntää myös suomen kielelle. Työhön kuului olennaisena osana myös vikakoodien indikoinnin parantaminen ja selkiyttäminen metsäkoneenkuljettajan kannalta.

Työ edellytti perehtymistä CAN-väylään ja sen toimintaan, Omronin valmistamaan ohjelmoitavaan logiikkakontrolleriin (PLC) sekä kosketusnäyttöön ja niiden ohjelmointiin. PLC:n ohjelmointi tapahtui Omronin omiin tuotteisiinsa kehittämällä automaatio-ohjelmalla, jolla ohjelmointi nojaa pääosin symbolimaiseen tikapuu-(ladder-)ohjelmointiin. Myös näytön ohjelmointiin oli oma ohjelmansa, jolla lähinnä luotiin näytölle tulevat kuvat ja painikkeet sekä yhdistettiin ne automaatio-ohjelmassa toteutettuun tikapuu-verkoston. Olennaista oli ymmärtää, kuinka PLC käsittelee CAN-viestit, ja miten ohjelma osaa yhdistää tietyn CAN-viestin tiettyyn tapahtumaan (esimerkiksi moottorin kierrosluku ja sen näyttäminen näytöllä).

Työn lopputuloksena saatiin paranneltu vikakoodien indikointi sekä suomeksi käännetyt EEM4-spesifikaation mukaiset vikakoodit. Työssä toteutetut parannukset tulevat käyttöön Profi 50 – harvestereissa.

## ABSTRACT

**Author:** Vartiainen Jouko

**Title of the Publication:** Sisu Engine Fault Code Indication and Translation

**Degree Title:** Bachelor of Engineering

**Keywords:** CAN-bus, forest machine automation, Omron

The aim of this bachelor thesis was to transmit the fault codes from the harvester's engine to the driver's awareness. ProfiPro Oy, who ordered this thesis, wanted to update the fault codes which are acquired via CAN-bus according to the Agco Power's newest EEM4-specification (electronic engine management). The specification works according to the SAE J1939 CAN-standard which is used by the heavy-duty vehicles. In addition, ProfiPro Oy wanted to translate the fault codes from English to Finnish and improve the fault code indication for the harvester driver.

The thesis required to be familiarized with the CAN-bus and its operations, programmable logic controller (PLC) and the touch-screen which are manufactured by Omron and with their programming. PLC's programming required a special program for automation programming, and it was also made by Omron. The programming method is a symbolic ladder programming. Omron provides also its own program for the touch-screen, and it is used to create layouts with different images and buttons, and connects those layouts to the ladder-program. It was vital for the thesis to understand how PLC handles CAN-messages and how it is capable to connect the specific CAN-message to the specific action (for instance engine's rpm and visualizing it to the touch-screen).

The result of this thesis was improved fault code indication and fault codes translated into Finnish according to the EEM4-specification. The improvements of this thesis will be used in Profi 50-harvesters in the future.

## SYMBOLILUETTELO

ACK	Acknowledgement bit
CAN	Controller Area Network
CAN_H	CAN High
CAN_L	CAN Low
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DATA	Datan määrä CAN-viestissä
DLC	Data Length Code
DSP	Display
ECU	Electronic Control Unit
EEM4	Sisu Diesel Stage 4 Electronic Engine Management
EOF	End Of Frame
FMI	Failure Mode Identification
ID	Identifier
IDE	Identifier Extension
IFS	Interframe Space
I/O	Input/Output
MIL	Malfunction Indicator Lamp
PGN	Parameter Group Number
PLC	Programmable Logic Controller

RTR	Remote Transmit Request
SAE	Society of Automotive Engineers
SOF	Start Of Frame
SPN	Suspect Parameter Number
SRR	Substitute Remote Request
μC	Microcontroller

## ALKUSANAT

Haluan kiittää ProfiPro Oy:tä työn tarjoamisesta, ja kehityspäällikkö Ari Mäenpäästä avusta ja neuvoista työn aikana kohdattujen ongelmien aikana.

## SISÄLLYS

1 JOHDANTO.....	1
2 PROFIPRO OY .....	2
3 AGCO POWER .....	3
4 CAN-VÄYLÄ.....	4
4.1 Viestien lähettäminen CAN-väylässä .....	6
4.1.1 Standardi CAN-kehys.....	6
4.1.2 Laajennettu CAN-kehys .....	8
4.2 CAN-väylä fyysisesti.....	8
4.3 SAE J1939 .....	9
4.4 PCAN-USB.....	10
5 OMRON CORPORATION .....	11
5.1 PLC-logiikka Omron SYSMAC CJ2M-CPU33 .....	11
5.2 Omron CJ1W-CORT21 .....	13
5.3 Näyttö Omron NS-CA002 .....	16
6 TYÖN SUORITTAMINEN.....	17
6.1 Logiikan yhdistäminen CAN-väylään.....	17
6.2 Indikoinnin luominen ja parantaminen CX-Programmer-ohjelmalla....	20
6.2.1 MIL-valon sytyttäminen (varoitusta) .....	21
6.2.2 Vilkkuva MIL-valo ja "Hätä Seis"-valo.....	25
6.3 CX-Programmerilla luodun indikoinnin siirtäminen näytölle CX- Designerilla .....	27
6.3.1 MIL-valo (varoitusta) .....	28
6.3.2 Vilkkuvat "Hätä Seis"- ja MIL-valot .....	29
6.4 Vikakoodien kääntäminen englannista suomeksi .....	33
7 TESTAUS JA POHDINTA .....	34
8 YHTEENVETO .....	35

LÄHTEET .....	36
---------------	----



## 1 JOHDANTO

Tämä insinöörityö on saatu ProfiPro Oy -nimiseltä harvesterivalmistajalta. Kajaanin ammattikorkeakoulu toimi välittäjänä insinöörityön saamisessa, ja kuultuani työn aiheen halusin tämän opinnäytetyön itselleni.

Insinöörityön tarkoituksena on välittää ProfiPro valmistamassa Profi 50 -harvesterissa käytettävän Agco Powerin Sisu-dieselmoottorin vikakoodit kuljettajan tietoisuuteen ja päivittää vikakoodit Agco Powerin uuden EEM4-spesifikaation mukaisiksi. Uudet vikakoodit tulee näkyä kuljettajalle sekä suomeksi että englanniksi. Pääpaino oli kuitenkin vikojen indikoinnin parantamisessa koneen näytöllä, jolta kuljettaja näkee harvesterin olennaiset tiedot, kuten esimerkiksi moottorin kierrosluvun ja polttoainemäärän. Näytölle luotiin vika-valot, jotka sekä ilmoittavat ilmenneestä viasta että kriittisestä viasta, jolloin harvesteri tulee pysäyttää välittömästi.

Työn tekeminen edellytti tutustumista CAN-väylään ja sen toimintaan. Moottorilta tulevat viestit etenevät CAN-väylässä, ja viestit välittyvät kuljettajan tietoon ohjelmoitavan logiikkakontrollerin (PLC) avulla. PLC toimii siis välikappaleena moottorin ja muiden CAN-väylään kytkettyjen laitteiden sekä kuljettajan välillä. PLC ja sen ohjelmoiminen olivat itselleni täysin uusia kokemuksia, joten laitteistoon ja ohjelmiin tutustuminen otti myös oman aikansa.

ProfiPro toimitti insinöörityön alkajaisiksi tarvittavat komponentit (logiikkakontrolleri, CAN-yksiköt, näytön) sekä lisenssillä varustetun ohjelman ohjelmointiin. Ongelmia tuotti PLC:n toiminnan testaaminen CAN-väylällä, sillä tietokoneen ja PLC:n välille tulevaan CAN-simulaatioon tarvittavan komponentin hankinta venyi aina tammikuun 2015 loppupuolelle, johtuen ongelmista tavarantoimittajan kanssa.

## 2 PROFIPRO OY

ProfiPro Oy on Profi-merkkisiä metsäkoneita valmistava sekä markkinoiva yritys, joka on perustettu vuonna 2009. Yritys toimii Pohjois-Pohjanmaalla Nivalassa, ja se keskittyy kehittämään harvestereita yhdessä asiakkaiden sekä alan erikoissektoreilla operoivien johtavien valmistajien kanssa.

Profi-harvesterit juontavat juurensa aina 80-luvun lopulle asti, jolloin koneita myytiin Nokka-harvestereina. Kaiken kaikkiaan harvestereita on valmistettu noin 650 kappaletta sekä Nokka- että Profi-nimillä. Nykyään ProfiPro Oy:n lippulaivana toimii vuonna 2005 lanseerattu Profi 50 -harvesteri, jonka ympärille tämä insinööritoiminta kiteytyy. [1.]

### Profi 50 -harvesteri

Profi 50 -harvesteri on suunniteltu metsätalouden pääte- ja harvennushakkuihin. Sen erikoisominaisuuksiin kuuluu PMC-pohjainen ohjaus- ja etähuoltojärjestelmä, jonka avulla konetta voidaan opastaa, huoltaa sekä ohjata tehtaalta käsin. PMC-automaatiikka on yhdistetty kookkaaseen 8-tuuman Omronin valmistamaan kosketusnäyttöön, jonka avulla kuljettaja voi määritellä esimerkiksi laitteen kahvat kunkin kuljettajan mieltymysten mukaisiksi, sekä mahdollisen komponenttien kuten kantokäsittely yms. laitteiden lisäys voidaan suorittaa etänä tehtaalta käsin, ilman että laitteita lähetettäisiin tehtaalle konfiguroitaviksi. Mekaaninen asennus lisäkomponenteille tapahtuu maastossa joko asiakkaan toimesta tai sitten tehtaalla huoltohenkilökunnan asentamana. Näytöllä näkyvät kaikki kuljettajalle tarpeelliset tiedot aina harvesterin mittalaitteesta moottorin kierroslukuun. Kaikki harvesterissa tapahtuvat asiat ovat ohjattavissa sekä seurattavissa näytön avulla.

PMC-automaatiikka ohjaa käytännössä kaikkia Profi 50 -harvesterin toimintoja. Sen avulla on korvattu erilliset ilmastoinnin, alustakoneen ja ajoautomaatiikan ohjaukset. Näiden lisäksi PLC-pohjaisella automaatiikalla ohjataan myös harvesterin nelisylinteristä Agco Powerin valmistamaa dieselmoottoria. [2.]

### 3 AGCO POWER

AGCO Power (entinen AGCO SISU POWER, Sisu Diesel) on Nokian Linnavuorella sijaitseva dieselmoottorivalmistaja. Yritys on fuusioitunut amerikkalaisen AGCO-konsernin kanssa, mikä tekee siitä yhden maailman suurimmista dieselmoottoreiden valmistajista [3]. AGCO Power valmistaa dieselmoottoreita useisiin maailman johtaviin traktori- ja maatalouskoneisiin, kuten esimerkiksi Valtralle [4] ja Massey Fergusonille [5]. Profi 50 -harvesteri käyttää AGCO Powerin valmistamaa Agco Sisu Powerin viidennen sukupolven nelisylinteristä Stage-4-diesel-moottoria.

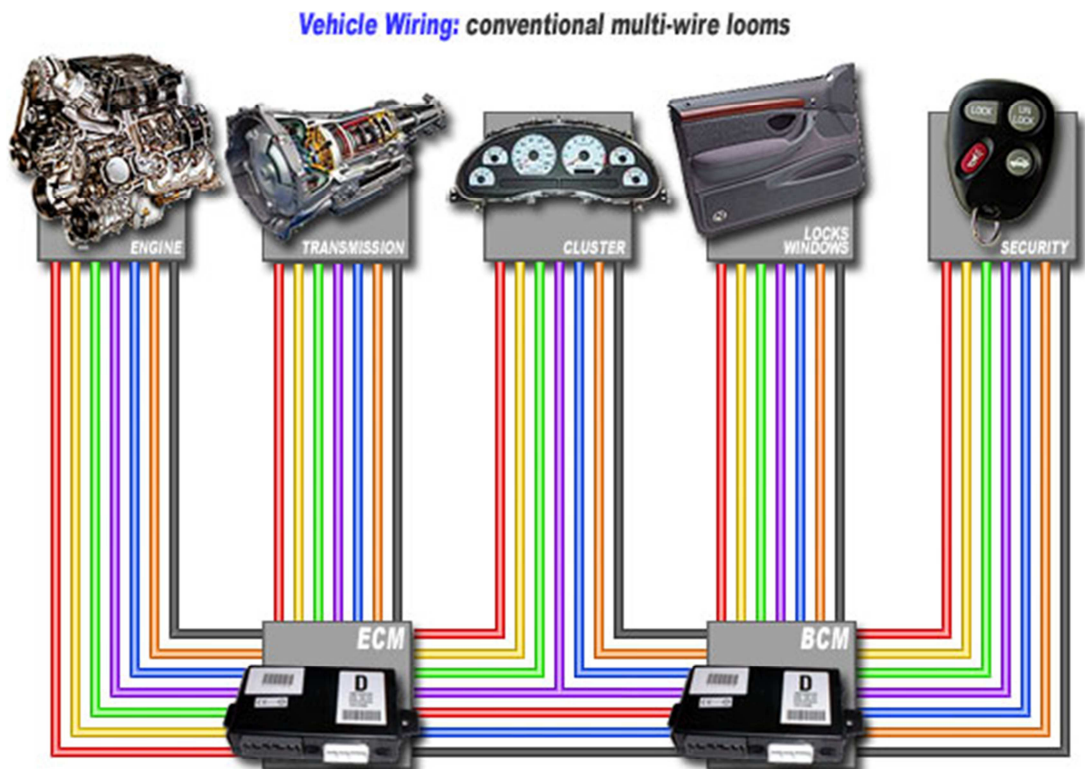
Agco Sisu Power, viidennen sukupolven Stage 4 -dieselmoottori

Profi 50 -harvesterissa käytetään turboahdettua Agco Sisu Powerin Stage-4 diesel-moottoria. Moottorin iskutilavuus on 4,9 litraa, ja siitä on otettu tehoa 140kW. Vääntömomentti on 700 Nm. Moottorin erikoisuus on sen pienehkö polttoaineen kulutus, ainoastaan alle 10 litraa tunnissa. [2.]

## 4 CAN-VÄYLÄ

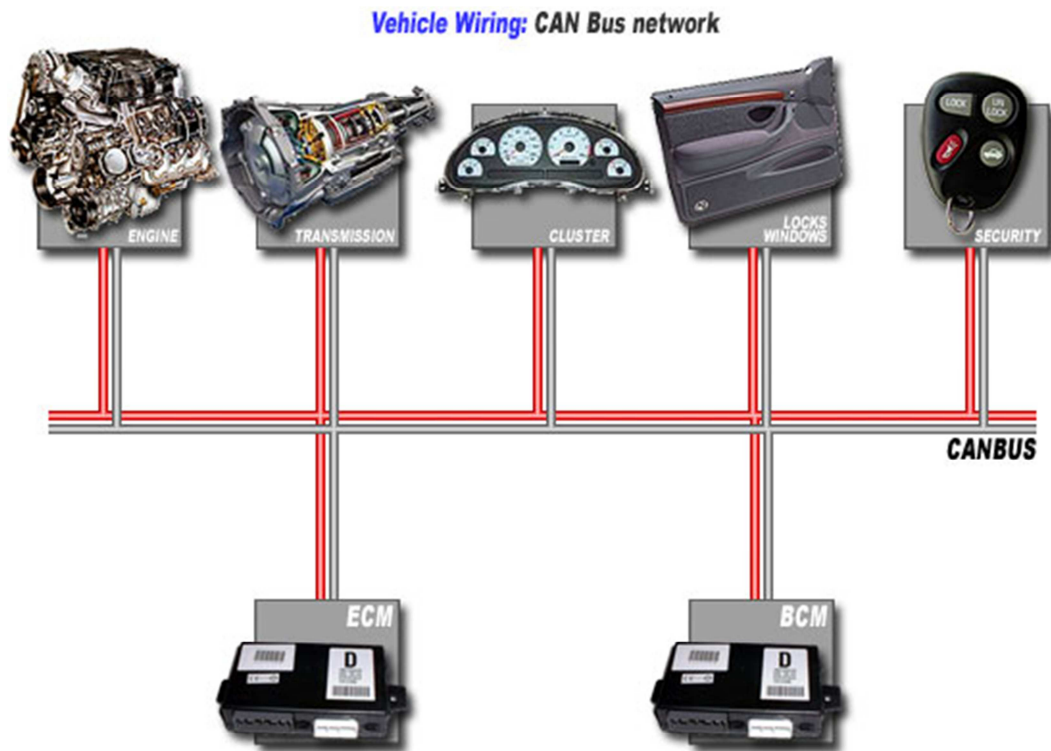
CAN-väylä (Controller Area Network) on Robert Boschin kehittämä sarjaliikenne protokolla, joka tukee reaaliaikaohjausta. Se on myös hyvin turvallinen tapa tiedonsiirrolle [6]. Turvallisuudesta, varmatoimisuudesta sekä helppokäyttöisyydestä johtuen CAN-väylää käytetään laajalti nykyaikaisissa ajoneuvoissa, joita varten se on ensisijaisesti kehitettykin. Samat syyt ovat myös vaikuttaneet siihen, että CAN-väylän käyttö on lisääntynyt oleellisesti muillakin teollisuuden aloilla sekä lääketeollisuuden puolella. [7.]

CAN-väylän etu ajoneuvoissa ja muissakin teollisissa järjestelmissä on se, että (sähkö)kaapeleiden määrä esimerkiksi autossa pienenee oleellisesti. Kun elektroniikan määrä ajoneuvojen kehittyessä kasvoi, ajoneuvot muuttuivat kaapeloinnin osalta monimutkaisemmiksi ja haavoittuvammiksi, kuten kuvasta 1 voidaan havaita. [6.]



Kuva 1. Ajoneuvon elektroniikka ilman CAN-väylää [6]

Kuten kuvasta 1 voidaan havaita, laitteiden yhdistäminen erillisillä kaapeleilla toisiinsa vaatii helpostikin useita (satoja) metrejä kaapeleita. Lisäksi mahdollisten vikojen ilmetessä diagnosointi olisi haastavaa, sillä kyseisellä tavalla rakennetun elektroniikan ymmärtäminen on hyvin haastavaa, varsinkin verrattuna vastaavaan laitteeseen toteutettuun CAN-järjestelmään, joka nähdään kuvassa 2. [6.]



Kuva 2. Ajoneuvon elektroniikka CAN-väylällä toteutettuna [6]

Kuten kuvasta 2 nähdään, CAN-väylä helpottaa oleellisesti ajoneuvon vikadiagnostiikkaa kuvaan 1 verrattuna. Jokaisen laitteen välillä parikaapelina välittyvä CAN-verkko, jolloin väylässä liikkuvat viestit lähetystä(transmit) ja vastaanottoa(receive) CAN-väylässä olevien laitteiden eli noodien välillä.

Ensimmäinen ajoneuvo, jossa käytettiin CAN-väylää, oli BMW 850 coupe vuonna 1986. Tarvittavien kaapeleiden määrä pieneni jopa noin kahdella kilometrillä, ja myös auton paino keveni noin 50:llä kilolla. Tämä mahdollisti erittäin nopean tiedonsiirron auton laitteiden välillä ja laitteiden kehittymisen entisestään, mutta kehitys myös teki laitteista itsestään monimutkaisempia [6].

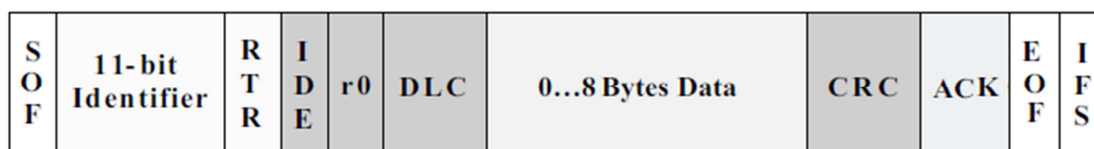
Siirtonopeus CAN-väylässä on kuitenkin täysin riippuvainen CAN-verkon pituudesta. Mikäli verkon pituus on alle 40 metriä, maksimisiirtonopeudeksi on määritetty 1 Mb/s. Pidemmällä matkoilla siirtonopeus heikkenee oleellisesti: 500 metriä pitkässä CAN-verkossa siirtonopeus on maksimissaan 125 kb/s, ja vastaavasti kilometrin pituisissa verkoissa siirtonopeus on enää 50 kb/s. [7.]

#### 4.1 Viestien lähettäminen CAN-väylässä

CAN-väylä on sarjaliikennejärjestelmä, jolla voi olla useita master-laitteita. Tällä tarkoitetaan sitä, että jokainen CAN-noodi eli väylään yhteydessä oleva laite voi lähettää väylään samanaikaisesti viestejä, ja osa voi myös vastaavasti vaatia tulevaa viestiä väylältä jatkuva-aikaisesti. Nämä viestit välittyvät laitteelta toiselle viestikehyksissä [7]. Viestit erotellaan viestin tunnistimen (ID) avulla ennalta ohjelmoituun prioriteettijärjestykseen pohjautuen. Viestit voidaan lähettää standardissa CAN-kehyksessä tai laajennetussa CAN-kehyksessä. Standardissa CAN-kehyksessä viestin ID on 11-bittinen, ja laajennetussa CAN-kehyksessä vastaavasti 29-bittinen. Käytännössä tämä tarkoittaa sitä, että standardilla CAN-kehyksellä voidaan lähettää 211 eli 2048 erilaisella ID-luvulla olevaa viestiä, kun laajennetulla kehyksellä voidaan lähettää 229 eli 537 miljoonaa erilaisella ID-luvulla olevaa viestiä.[8.]

##### 4.1.1 Standardi CAN-kehys

Standardi CAN-kehys, jossa on 11-bittinen ID, on kuvan 3 mukainen.



Kuva 3. Standardi 11-bittinen CAN-kehys [8, s.3]

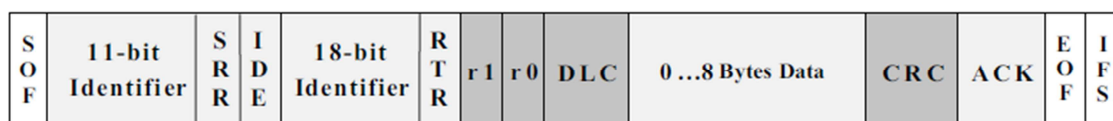
Kuten kuvasta 3 nähdään, standardi CAN-kehys sisältää useita osioita tunnistimen lisäksi. Nämä osiot on luetteloitu seuraavaksi järjestyksessään vasemmalta oikealle.

- SOF: Start Of Frame, yksittäinen kehyksen aloitusbitti, jota käytetään myös noodien synkronoinnissa, kun väylä on kytketty päälle.
- Identifier: ID, standardin CAN-kehyksen 11 tunnistinbittiä, jotka määrittävät viestin prioriteettijärjestyksen – mitä pienempi binaariarvo on, sitä korkeampi on viestin prioriteetti.
- RTR: Remote Transmission Request, etäsiirtopyyntöbitti, joka on dominoivassa asemassa, mikäli toisen CAN-noodin tietoa tarvitaan. Kaikki väylän noodit saavat viestin, mutta ID-luku määrittää viestin varsinaisen kohteen. Vastaavasti pyynnön vastauksena tuleva tieto on myös kaikkien väylään kytkettyjen noodien saatavilla.
- IDE: Identifier Extension, tunnistimen laajennusbitti. Määrittää käytettävän kehyksen (standardi tai laajennettu CAN-kehys).
- r0: käytetään, mikäli joskus halutaan laajentaa standardia.
- DLC: Data Length Code, 4-bittinen datan pituuskoodi, jonka avulla määritetään lähetettävän datan pituus.
- Data: Lähetettävä viestin varsinainen data, joka on pituudeltaan maksimissaan 64 bittiä.
- CRC: Cyclic Redundancy Check, 16-bittinen alue, jolla lähetetään tarkistussumma, jonka avulla voidaan todeta vastaanotetun datan virheettömyys.
- ACK: Acknowledgement, tiedostusbitti. Vastaanotettuaan viestin vastaanottaja kuittaa saapuneen viestin pakottamalla tiedostusbitin 1-tilaan.
- EOF: End Of Frame, kehyksen 7-bittinen lopetusalue, joka lopettaa CAN-kehyksen ja pysäyttää bittivirran.

- IFS: Interframe Space, kehysten välinen 7-bittinen avaruus, joka sisältää kontrollerin vaatiman ajan saadun kehyksen liikuttamiseksi oikeaan muistipaikkaan.[8.]

#### 4.1.2 Laajennettu CAN-kehys

Laajennettu CAN-kehys on kuvan 4 mukainen.



Kuva 4. Laajennettu 29-bittinen CAN-kehys [8, s.4]

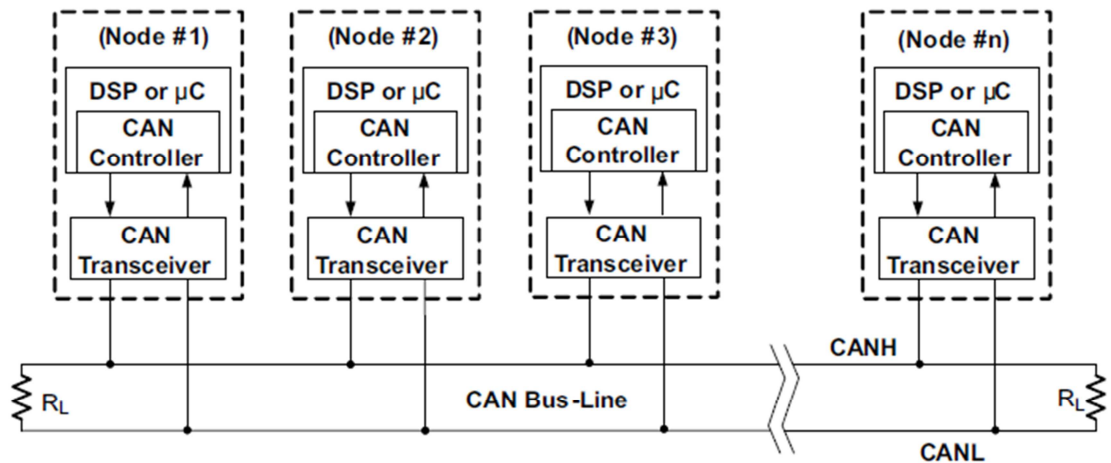
Laajennettu 29-bittinen CAN-kehys poikkeaa hieman 11-bittisestä standardista CAN-kehyksestä, ja nämä erot käydään läpi seuraavaksi.

- SRR: Substitute Remote Request, korvaava etäpyyntöbitti, joka korvaa RTR:n laajennettussa CAN-kehyksessä.
- IDE: Resessiivinen bitti tunnisteiden laajennusbitissä merkitsee sitä, että tämän laajennusbitin jälkeen seuraa 18 tunniste-bittiä.
- r1: Lisätty reservibitti. [8.]

#### 4.2 CAN-väylä fyysisesti

Fyysisesti CAN-väylä koostuu kahdesta kaapelista, joista toinen on CAN\_H ja toinen CAN\_L. Ne ovat tasapainoisia toisiinsa nähden, mutta virran suunta johtimien välillä on eri. CAN\_H- ja CAN\_L-johtimet on yhdistetty molemmista päistä 120  $\Omega$  vastuksella, joiden avulla estetään mahdolliset signaalin heijastumiset. CAN-väylän fyysinen rakenne on kuvan 5 kaltainen. [8.]





Kuva 5. CAN-väylän fyysinen rakenne [8, s.7]

CAN-väylän looginen 1- ja 0-tila saadaan selville CAN\_H- ja CAN\_L-johtimien välisestä jännite-erosta, jolloin siitä voidaan käyttää nimeä differentiaalinen väylä. Tällä saavutettu käytännön etu on se, että ulkoisia häiriötä saadaan pienennettyä varsinaisesta signaalista oleellisesti, sillä ne vaikuttavat sekä CAN\_H- että CAN\_L-johtimiin. Ollessaan resessiivisessä, niin sanotussa uinuvassa tilassa, sekä CAN\_H että CAN\_L ovat jännitearvoltaan noin 2,5 voltissa. Vastaavasti väylän dominantissa tilassa CAN\_H kasvaa voltilla 3,5 volttiin, ja vastaavasti CAN\_L laskee voltilla 1,5 volttiin, luoden näin 2-volttisen differentiaalisignaalin. [8.]

#### 4.3 SAE J1939

SAE J1939 on ajoneuvovinsinöörien järjestön (SAE) kehittämä CAN-standardi, jota käytetään eri teollisuuden aloilla, esimerkiksi maa- ja metsätalouden koneissa. J1939 on korkeatasoinen CAN-standardi, joka määrittelee, kuinka väylässä tapahtuva kommunikointi eri noodien eli moduulien välillä tapahtuu. J1939-pohjainen CAN-väylä on hyvin spesifinen järjestelmä, eikä sitä voida juurikaan yleistää muihin erilaisiin CAN-standardeihin. J1939-standardi käyttää viestinnässään laajennettua 29-bittisellä ID-luvulla olevaa CAN-kehystä, jossa määritetään viestin prioriteetti, viestin lähettänyt ajotietokoneyksikkö (ECU) sekä

8-bittinen data. ECU tunnistaa saapuneen viestin datatyypin parametrinen ryhmänumeron (PGN) avulla. [9.]

#### 4.4 PCAN-USB

PCAN-USB on Peak-Systemsin kehittämä yksinkertainen adapteri, jolla voidaan yhdistää CAN-väylä tietokoneen USB-porttiin. Adapterin ja erillisen Windows-käyttöjärjestelmän kanssa yhteensopivan PCAN-View-monitorointiohjelman avulla voidaan sekä lähettää että vastaanottaa viestejä CAN-väylästä. Adapteri on kuvan 6 mukainen. [10.]



Kuva 6. PCAN-USB-adapteri [10]

Kyseisiä välineitä käytettiin Profi 50 -harvesterin ECU:n simulointiin lähettämällä Sisun spesifikaatiossa löytyviä CAN-viestejä. Tällä tavalla pystyttiin tutkimaan PLC-logiikan toimintaa tavalla, joka vastaisi varsinaista harvesterissa käytettävää tuotetta.

## 5 OMRON CORPORATION

OMRON Corporation (alunperin Tateisi Electric Manufacturing Company) on japanilainen vuonna 1933 perustettu elektroniikka-alan yritys, joka on erikoistunut automaatiotekniikkaan teollisuudessa [11]. Automaatiotekniikan ohella Omron kehittää komponentteja elektroniikka- ja autoteollisuudelle, tuotteita ja palveluja sosiaalisten järjestelmien (esimerkiksi liikenne, turvallisuus) ja terveydenhuollon aloille. [12.]

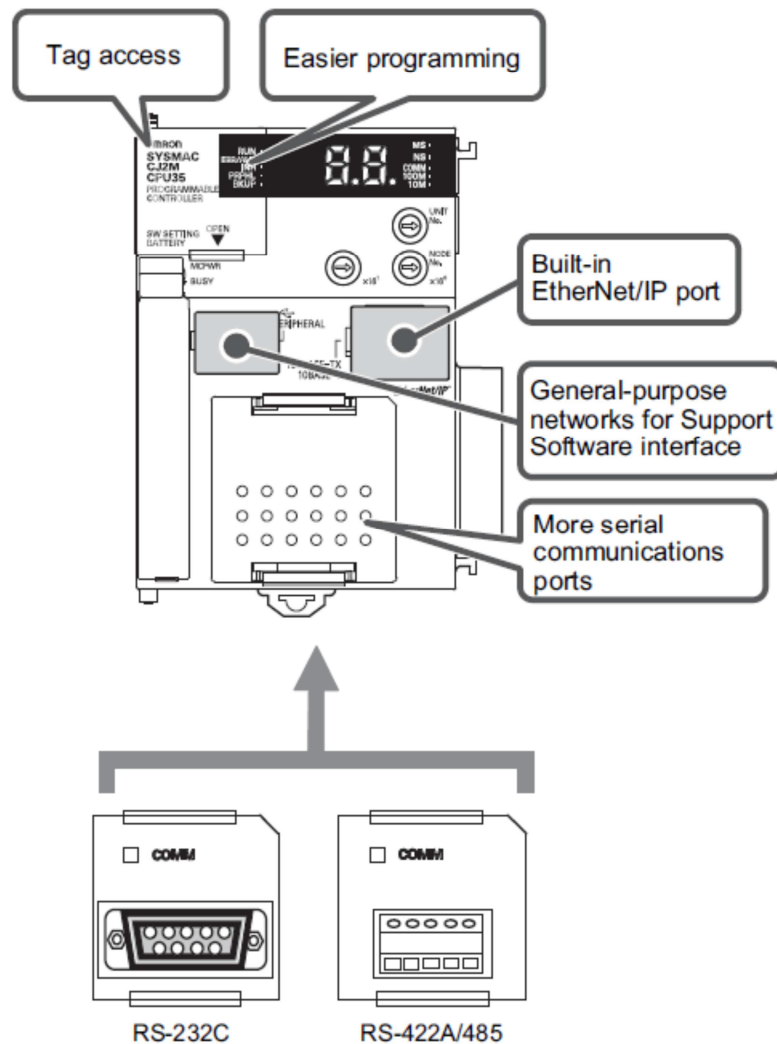
Profi 50 -harvesterissa käytetään Omronin valmistamaa PLC-logiikkaa. Tämä logiikka toimii välikappaleena sekä AGCO Powerin moottorin että muiden harvesterissa olevien CAN-ecujen välillä ja välittää tiedot kuljettajalle hytissä olevan näytön avulla.

### 5.1 PLC-logiikka Omron SYSMAC CJ2M-CPU33

SYSMAC CJ2M-CPU33 on Omronin valmistama ohjelmoitava logiikkakontrolleri (PLC). Se on logiikkaverkon suoritin (CPU), jolla on useita eri liitäntöjä, kuten esimerkiksi Ethernet-liitäntä (logiikan sisäinen, esimerkiksi CPU33:n ja NS-CA002-näytön välillä), RS232C- ja RS-422A-liitännät sekä yleiset USB- ja Ethernet-verkkoliittimet ohjelmistokäyttöliittymää varten. Kuvassa 7 nähdään logiikkakontrolleri ja sen liitännät. [13.]

## CJ2M CPU Units

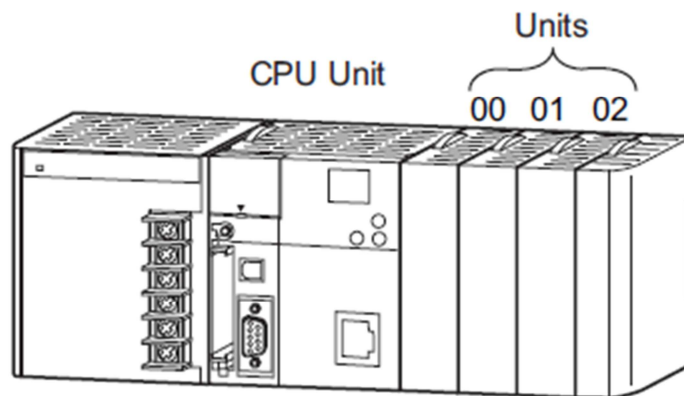
### Standard CPU Units (CJ2M-CPU3□)



Kuva 7. CJ2M-CPU33-logiikkakontrolleri ja sen liitännät [13, s. 49]

Omronin PLC-logiikka koostuu eräänlaisesta palapelistä, joka muodostetaan kuvan 8 mukaisella tavalla. Tähän palapeliin liittyy olennaisesti virtalähde, CJ2M-CPU33 ja siihen liitettävät muut yksiköt. Profi 50 -harvesterissa prosessoriin on kytketty kaksi CORT-21 CAN-yksikköä sekä NS-CA002-näyttö. Kuvassa 8 CPU Unitin puolella virtalähde, ja oikealla puolella prosessoriin liitettävät yksiköt (Units). Liitettävät voidaan asettaa löydettäväksi joko manuaalisesti tai automaattisesti. Automaattisella tavalla, jota myös Profi 50 -harvesterissa käytetään, tulo- ja lähtömuisti (I/O memory) katsoo prosessoriin kytketyt laitteet ja järjestee ne kyt-kemisjärjestyksen mukaan. Yksikköjen jälkeen asetetaan

päätylevy viimeisen yksikön kylkeen, jolloin prosessori ymmärtää lopettaa yksikköjen etsimisen. [13.]



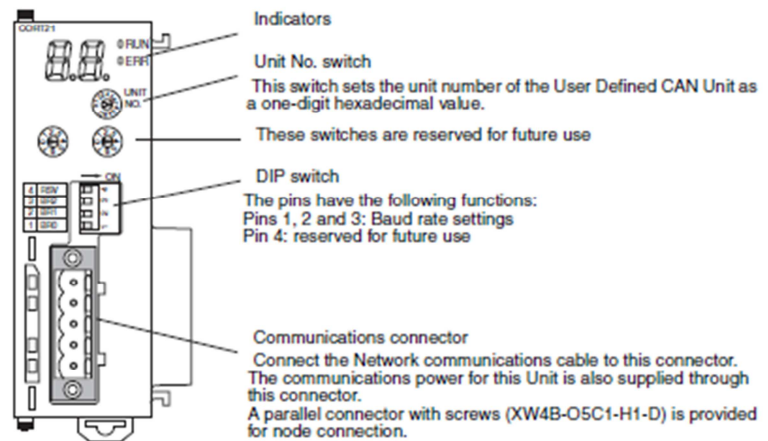
Kuva 8. PLC:n kokoaminen [13, s. 86]

Logiikkakontrollerin ohjelmointi tapahtuu CX-Programmer-ohjelmalla, joka on Omronin kehittämä ohjelmointiohjelmisto. Ohjelma tukee kaikkia Omronin kehittämiä PLC-tuotteita viimeisen 20 vuoden ajalta. Ohjelmointi tapahtuu yksinkertaisen tikapuu-(Ladder-)editorin avulla. [14.]

## 5.2 Omron CJ1W-CORT21

CJ1W-CORT21 on kuvan 9 mukainen käyttäjän määrittämä CPU-väyläyksikkö, joka mahdollistaa logiikkakontrollerin yhdistämisen CAN-väylään. Väyläyksikkö yhdistetään logiikkakontrolleriin osiossa 5.1 olevan kuvan 8 mukaisella tavalla. CPU-väyläyksikköjä voidaan yhdistää logiikkakontrolleriin yhteensä 16 kappaletta, ja yksikköjärjestys määritetään CORT21-yksikössä olevan säätimen avulla (Unit No. Switch), joka nähdään kuvassa 9. Laitteiden järjestysnumerot menevät siten, että 0 vastaa laitetta 1 ja 15 laitetta 16. [15.]

## CJ1W-CORT21



Kuva 9. CJ1W-CORT21-väyläyksikkö [15, s. 26]

Väyläyksikössä olevien dippikytkimien avulla (DIP-switches) asetetaan väylän tiedonsiirto-nopeus (Baud rate). Nopeus asetetaan kuvan 10 dippikytkimillä 1, 2 ja 3. Dippikytkin 4 ei ole käytössä. [15.]

## DIP Switch



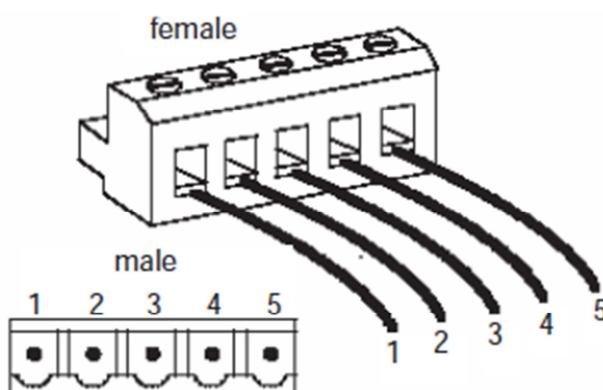
Kuva 10. CJ1W-CORT21-väyläyksikön dippikytkimet [15, s. 43]

Taulukossa avulla nähdään kuvan 10 dippikytkimillä määritetty väylän tiedonsiirtonopeus. Profi 50 -harvesterissa tiedonsiirtonopeus on 250 kbit/s.

Taulukko 1. CJ1W-CORT21-väyläyksikön mahdolliset tiedonsiirtonopeudet [15, s. 43]

DIP-switch			Baud rate (kbit/s)	Sample point (%)
1	2	3		
OFF	OFF	OFF	10	80
ON	OFF	OFF	20	80
OFF	ON	OFF	50	80
ON	ON	OFF	125	80
OFF	OFF	ON	250	80
ON	OFF	ON	500	80
OFF	ON	ON	reserved	reserved
ON	ON	ON	1000	70 <sup>1</sup>

Väyläyksikkö yhdistetään CAN-väylään kuvan 11 mukaisen 5-pinnisen avoliittimen avulla. Avoliittimen paikka on kuvan 9 mukaisesti väylän etupuolella, ja laitteeseen yhdistetään liittimen urospuoli (male). [15.]



Kuva 11. 5-pinninen avoliitin CAN-väylän liittämiseen CORT21-yksikköön [15, s. 43]

Naarasliittimeen (female) liitetty numeroidut kaapelit kuvaavat väylän signaaleja, jotka on selitetty taulukossa 2.

Taulukko 2. Avoliittimen signaalien selitykset [15, s. 44]

Pin no.	Signal	Color	Description
1	CAN_GND	Black	Ground
2	CAN_L	Blue	Communications line, dominant low
3	CAN_SHLD	----	Shield, optional
4	CAN_H	White	Communications line, dominant high
5	CAN_V+	Red	External positive power supply 24V, for powering the transceiver and optocouplers

Pinniin numero 1 syötetään väylän maa, ja pinniin 5 väylän käyttöjännite, joka on 24 voltia. CAN\_L- ja CAN\_H-signaalit syötetään pinneihin 2 ja 4, ja pinniin 3 tuleva suojasignaali on vaihtoehtoinen. [15.]

### 5.3 Näyttö Omron NS-CA002

Profi 50 -harvesterissa käytetään Omronin valmistamaa NS-CA002-kosketusnäyttöä. Profi Pro käyttää sekä 12- että 8-tuumaisia versioita näytöstä, ja näytön avulla kuljettaja pystyy seuraamaan koneen PLC-logiikassa tapahtuvaa viestintää. Näytön tehtävänä on välittää koneen tiedot kuljettajalle ja toimia käyttöliittymänä koneen ja kuljettajan välillä, jolloin näytölle välittyy reaaliajassa esimerkiksi moottorin kierrosluku, polttoaineen määrä, ajotunnit, mahdolliset vikakoodit ja niin edelleen. Profi 50 -harvesterissa NS-CA002 on kytketty logiikkakontrolleriin (CJ2M-CPU33) Ethernet-portin kautta.

NS-CA002-kosketusnäyttöä ohjelmoidaan Omronin kehittämällä CX-Designer-ohjelmalla, joka on integroitu CX-One-ohjelmistopakettiin [16]. Ohjelmalla luodaan näytöllä olevat kuvakkeet, joille luodaan aina oma pohjansa (esimerkiksi alkunäkymä, jossa on koneen mittaristo ja painikkeet valokatkaisijoille, josta pääsee uuteen näkymään). Näihin pohjiin voidaan lisätä graafisia elementtejä, ponnahdusikkunoita sekä painikkeita, joilla voidaan tehdä joko valintoja (valot päälle/pois) tai siirtyä uuteen ikkunaan. CX-Designer-ohjelmalla tapahtuva ohjelmointi ei edellytä varsinaista koodaamista, vaan pohjalle valitaan kuvake erillisestä valikosta. Käyttäjän tarvitsee määrittää ainoastaan kuvakkeen koko, muoto sekä se, mikä vaikuttaa kuvakkeen aktivoitumiseen.



## 6 TYÖN SUORITTAMINEN

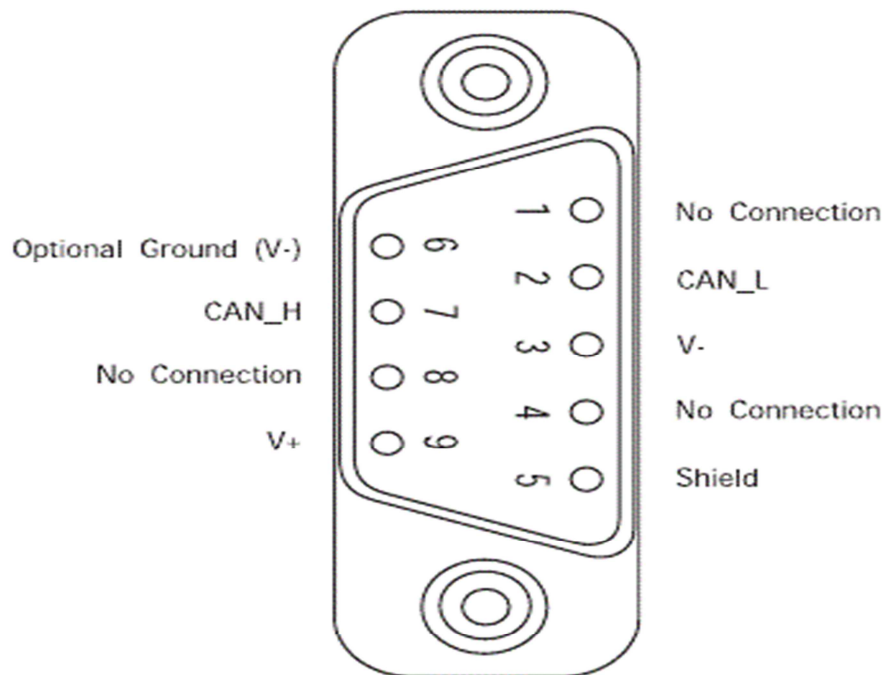
Insinööriyön tarkoituksena on siis välittää Sisu-moottorilta saatavat vikakoodit PLC-logiikkaan ja siitä edelleen kuljettajan tietoisuuteen. Alun perin oletuksena oli, että vikakoodit täytyy kysyä moottorilta erillisellä CAN-viestillä, jolloin moottori lähettää listan vikakoodeista. Tämä kuitenkin osoittautui turhaksi tämän insinööriyön kannalta, sillä Agco Powerilta saadun dokumentaation perusteella moottori lähettää joka tapauksessa CAN-viestin uuden vian ilmetessä. Koska PLC on ohjelmoitu siten, että viat tallentuvat logiikan muistiin ja niistä tuoreimmat näkyvät kuljettajalle, ei vikakoodilistan kyselyä koettu tarpeelliseksi.

Edellisestä johtuen työn varsinaiseksi tavoitteeksi muotoutui vikaindikoinnin parantaminen ja selkiyttäminen kuljettajan kannalta. Tämän lisäksi vikakoodit haluttiin päivittää aiemmasta EEM3-spesifikaatiosta nykyisen EEM4-spesifikaation mukaiseksi, ja vikakoodit haluttiin kääntää englannista suomeksi.

### 6.1 Logiikan yhdistäminen CAN-väylään

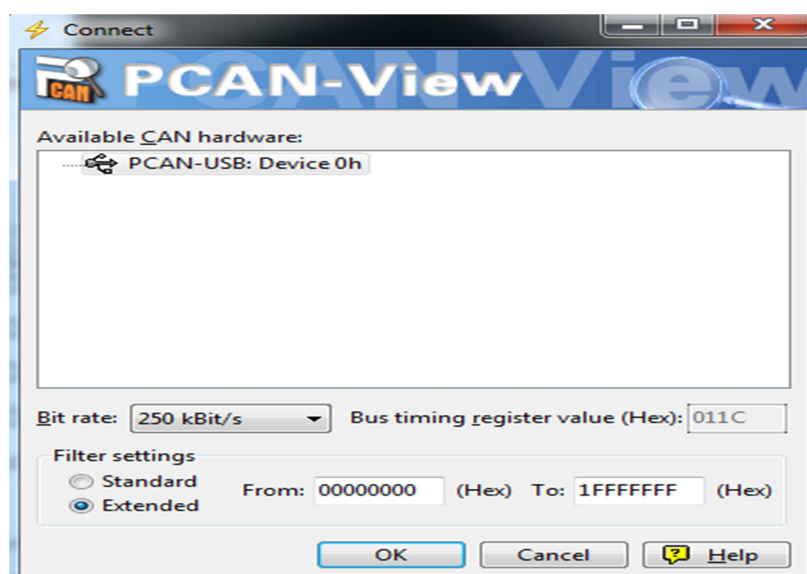
PLC yhdistetään CAN-väylään CORT21-yksikön avulla. CAN-väylää simuloitiin PCAN-USB:n avulla, jolloin tietokoneella käytettävän PCAN View -ohjelman avulla voitiin hallinnoida CAN-väylää. Ohjelman avulla on mahdollista sekä lukea että lähettää CAN-viestejä.

Fyysisesti PCAN-USB yhdistettiin CORT21-yksikköön kuvien 11 ja 12 mukaisesti. Ulkoisesta virtalähteestä otettu 24-volttinen käyttöjännite ja maa kytkettiin pinneihin 5 ja 1, sekä CAN\_L ja CAN\_H kytkettiin pinneihin 2 ja 4. CAN\_L ja CAN\_H pinnien väliin asetettiin myös vaadittu 120  $\Omega$  päätevastus. Pinni 3 eli CAN\_SHLD jätettiin kokonaan kytkemättä. CORT21-avoliittimeen menevät CAN\_L- ja CAN\_H-kaapelit yhdistettiin PCAN-USB-adapteriin kuvan 13 mukaisen d9-liittimen avulla. PCAN-USB-adapteriin kytkettiin ainoastaan pinnit 2 (CAN\_L) ja 6 (CAN\_H).



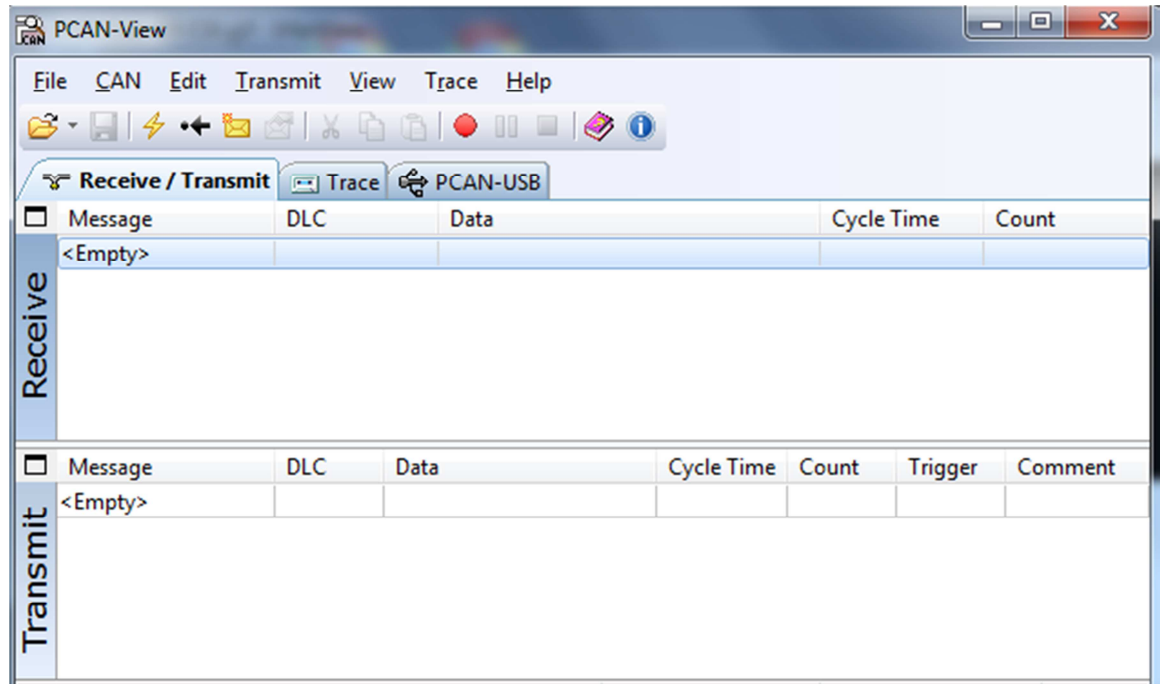
Kuva 12. D9-liitin PCAN-USB-adapteria varten [17]

PCAN-USB-adapterin käyttäminen edellyttää PCAN View -ohjelman asentamista. Ohjelma toimitetaan ajureineen adapterin mukana, mutta se on myös ladattavissa Peak Systemsin sivuilta [10]. Kun ohjelma, ajurit ja laite on asennettu, PCAN View on toimintavalmis. Kun PCAN View avataan asennuksen jälkeen, aukeaa kuvan 13 mukainen ikkuna.



Kuva 13. PCAN-View'n yhdistämisikkuna

Kuvassa 13 nähdään PCAN-View'n alkuikkuna. Ikkunasta nähdään löydetyt Peak Systemsin CAN-laitteet ja määritetään väylän asetukset. Väylän nopeudeksi asetettiin Profi 50 -harvesterissa käytettävä 250 kbit/s nopeus sekä laajennettu (extended) 29-bittinen CAN-kehys. OK-painikkeen napsauttamisen jälkeen aukeaa kuvan 14 mukainen terminaali-ikkuna.



Kuva 14. PCAN-View'n valvontaikkuna

Kuvassa 14 on valvontaikkuna, jonka avulla varsinaista väylää valvotaan. Receive-alue tarkoittaa CAN-väylää pitkin PCAN-USB:lle tulevia viestejä, ja Transmit-alueella voidaan luoda ja lähettää viestejä väylään. Uusi viesti luodaan kaksoisnapsauttamalla Empty-saraketta tai painamalla näppäimistön Insert-näppäintä, jolloin kuvan 15 kaltainen viestinluonti-ikkuna avautuu.

New Transmit Message

ID (Hex): 000 DLC: 8 Data: (Hex) 00 00 00 00 00 00 00 00 00 00

Cycle Time: 0 ms

☐ Paused

Message Type

☐ Extended Frame

☐ Remote Request

Comment:

OK Cancel Help

Kuva 15. Uuden lähetettävän CAN-viestin luominen

Kuvan 15 viestinluonti-ikkunassa CAN-viestille määritetään tunniste (ID), datan pituus (DLC) sekä varsinainen data. Viestin toistuvuus eli cycle time on määritetty Agco Powerin dokumentaatiossa viestikohtaisesti, eli esimerkiksi polttoaineen määrä -signaali lähetetään logiikalle sekunnin (1000 ms) välein. Viestin tyyppi (message type) on aina laajennettu (Extended). OK-painiketta painamalla viesti tallennetaan kuvan 14 Transmit-kenttään, josta sitä voidaan lähettää väylään aina haluttaessa.

## 6.2 Indikoinnin luominen ja parantaminen CX-Programmer-ohjelmalla

CX-Programmer-ohjelmassa varsinainen ohjelmointi tapahtuu tikapuu-editorin avulla. PLC:n viestiliikenne toimii hyvin pitkälti muistipaikkojen avulla, mikä tarkoittaa sitä, että esimerkiksi polttoaineen määrästä kertova CAN-viesti tallentuu tiettyyn muistipaikkaan. Tämän viestin tietyt bitit määrittävät tankissa olevan polttoaineen määrän, ja näitä bittejä luetaan koko ajan polttoainetta tarkkailevan funktioblokin avulla. Tämä funktioblokki toimii tietyssä koodaajan määrittämässä osoitteessa, jolloin CX-designerin avulla luotava polttoaineen

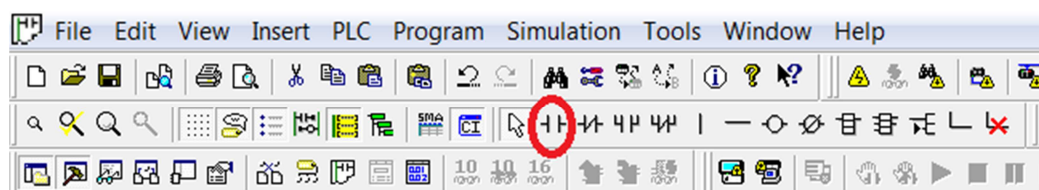
määrää kuvaava mittari lukee tiedon tästä osoitteesta ja näyttää saadun bittiarvon mukaisen polttoainemäärän näytöllä.

Vastaavalla tavalla toteutettiin myös viasta indikoivan MIL-valon (Malfunction Indicator Lamp) syttyminen. Kun vikakoodi tulee moottorilta logiikalle, datassa on mukana bitti, joka määrittää MIL-valon ja muiden viasta indikoivien valojen syttymisen. EEM4-spesifikaation mukaisesti viestissä määritetään myös erillisen keltaisen ja punaisen varoitusvalon syttyminen. Kun kriittinen eli punainen vika ilmenee, sekä punaisesta valosta että MIL-valosta haluttiin vilkkuvat kuljettajan huomion kiinnittämiseksi.

### 6.2.1 MIL-valon sytyttäminen (varoitusta)

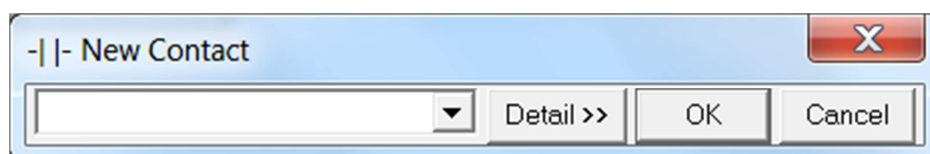
Jotta MIL-valo voitiin luoda, täytyi tikapuuverkkoon asettaa uusi poikkipuu eli rung. Valo sijoitettiin samaan otsakkeen alle, jossa määritellään vikakoodien lukeminen. Vikakoodien ID:t tulevat moottorilta PLC:n CIO-muistiin, jonka osoite on 5835, ja viestissä olevan datan ensimmäinen tavu tulee seuraavaan muistipaikkaan 5836. EEM4-spesifikaatiosta saatiin selville, että vikakoodin mukana määritetään viasta syttyvät valot, ja ne kerrotaan ensimmäisessä tavussa. Yhteen tavuun mahtuu dataa 8 bittiä (0–FF<sub>H</sub>), ja niillä ohjataan keltaista varoitusvaloa (bitit 4 ja 3), punaista varoitusvaloa (bitit 6 ja 5) sekä MIL-valoa (bitit 8 ja 7). Valojen syttymisen ja sammumisen kannalta pariton bitti on merkitsevämpi, sillä sen arvo määrittää, palaako valo vai ei (bitin arvo = 0, ei pala; bitin arvo = 1, palaa).

Tätä tietoa hyödyntäen pystyttiin suunnittelemaan ja luomaan valojen sytytysmetodi tikapuu-verkostoon. Uusi rung aloitettiin valitsemalla CX-Programmerin työkalupalkista uusi kontakti (new contact)-blokki kuvan 16 mukaisesti. Kuvassa 16 on blokin valinta korostettu punaisella.



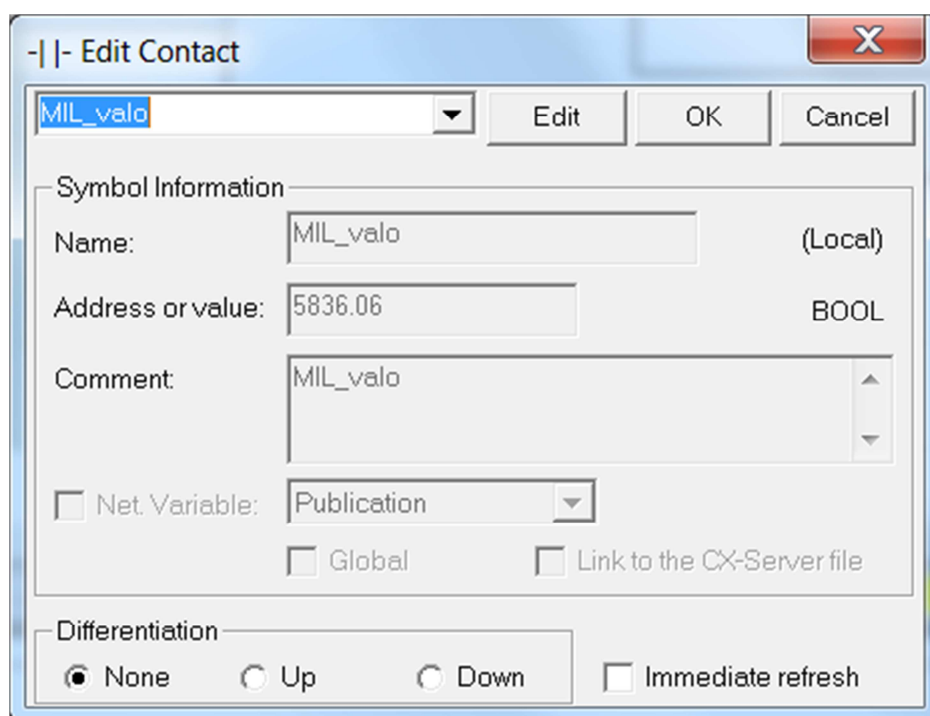
Kuva 16. New Contact -blokin valinta

Tämän jälkeen osoitin siirrettiin alueella, jossa tikapuuverkkoa luodaan. Kuten edellä todettiin, valojen sytytys luotiin aloittamalla uusi rung heti vikakoodien käsittelyn jälkeen. Kun osoitinta napsautettiin uuden rungin kohdalla, aukesi oheinen kuvan 17 mukainen ikkuna.



Kuva 17. New Contact -blokin luominen

Kuvan 17 ikkunassa voidaan luoda jo aiemmin luotu kontakti painamalla nuolesta, jolloin aukeaa lista jo aiemmin luoduista kontakteista. Koska nyt haluttiin luoda täysin uusi kontakti, painettiin nappia "Detail >>" ja määritettiin uudelle kontaktille tiedot kuvan 18 mukaisesti.



Kuva 18. Luodun kontaktiblokin määrittäminen

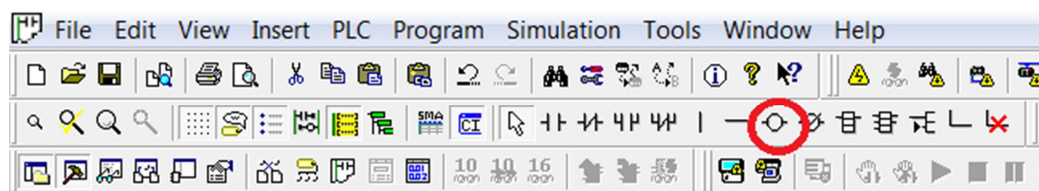
Kuvassa 18 määritettiin nimi symbolille (name), osoite (address) sekä kommentti (comment). Osoite määräytyi siten, että vikaviestin ensimmäinen tavu tulee

muistipaikkaan 5836, ja MIL-valo sytytetään viestin 7 bitillä. Kun halutaan viitata tiettyyn bittiin, osoitteen jälkeen laitetaan piste. CX-Programmerissa bitit 1–8 vastaavat numeroita 0–7. Tästä syystä MIL-valon osoitteeksi määritettiin 5836.06, eli muistipaikkaan tulevan 5836 tulevan tavun 7. bitti. Painamalla OK-painiketta luotu kontakti ilmestyy tikapuuverkkoon kuvan 19 mukaisesti.



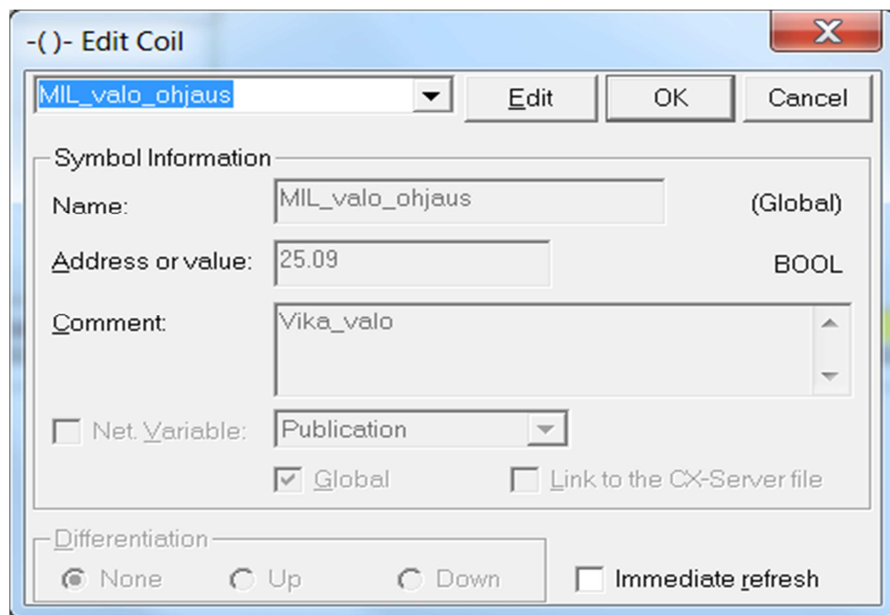
Kuva 19. MIL-valo kontaktiblokki tikapuuverkossa

MIL-valon kanssa haluttiin syttyvän myös vastaavalla tavalla keltaisen varoitusvalon, jota ohjataan bitillä 3. Osoite varoitusvalolle on siis 5836.02. Jotta MIL-valo saataisiin varsinaisesti syttymään näytölle, luotiin rungin loppuun silmukka (coil). CX-Designerilla luotu kuljettajalle näkyvä MIL-valo lukee silmukassa määritettyä osoitetta. Uusi silmukka luodaan valitsemalla New Coil -blokki CX-Programmerin työkalupalkista kuvan 20 mukaisesti.



Kuva 20. Uuden silmukka-blokin luominen

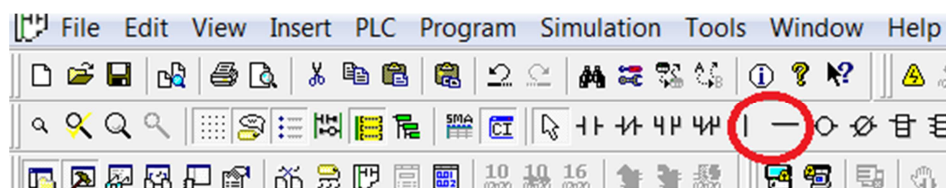
Valinnan jälkeen silmukan luonti onnistuu samalla tavalla kuin uuden kontaktin luominen. Osoitin viedään tikapuuverkkoon, klikkaus ja painamalla avautuneesta ikkunasta jälleen "Details >>"-painiketta, päästään käsiksi silmukan asetuksiin, jotka ovat kuvan 21 mukaiset.



Kuva 21. Luodun silmukan asetusten määrittäminen

Kuvassa 22 nähdään jälleen silmukan nimi, osoite ja kommentti. Osoitteeksi valikoitui 25.09, sillä mittariston indikoinnit on pääpiirteissään hoidettu osoitteilla 24 ja 25. 25.09 oli vapaana, joten tämä valikoitui MIL-varoitussalon osoitteeksi.

Luodut blokit yhdistettiin tikapuuverkossa toisiinsa pysty- ja vaakapuu-blokin (vertical, horizontal) avulla, joka valitaan käyttöön painamalla työkalupalkista New Vertical/Horizontal-blokkia kuvan 22 mukaisesti.



Kuva 22: Tikapuun valinta

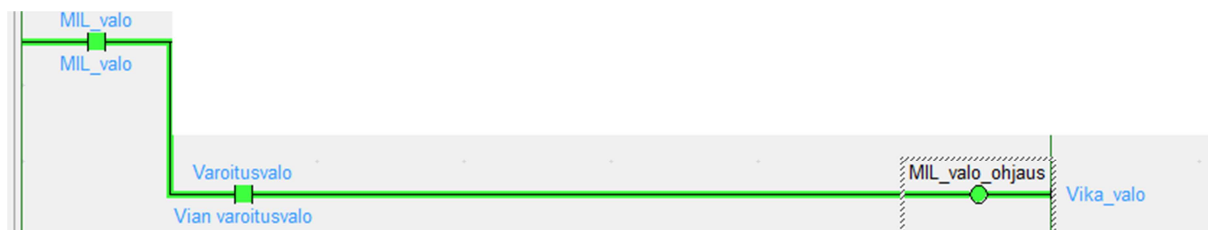
Blokit yhdistettiin verkossa toisiinsa, jolloin syntyi kuvan 23 mukainen poikkipuu.



Kuva 23. MIL-varoitussalon indikointi



Valo näytöllä syttyy, kun MIL\_valo\_ohjaus-silmukka menee ON-tilaan. Tämä edellyttää sitä, että vikakoodissa tulee MIL-valoa sekä varoitusvaloa ohjaava bitti. Kun valo on aktiivinen, rung näyttää kuvan 24 mukaiselta.

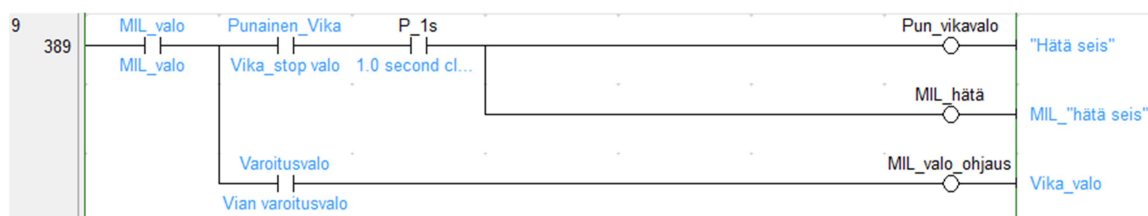


Kuva 24. Rung, kun MIL-valo on syttynyt näytöllä

### 6.2.2 Vilkkuva MIL-valo ja "Hätä Seis"-valo

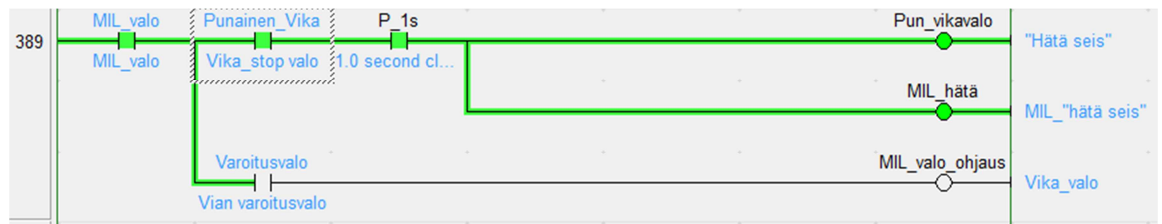
Vilkkuvan MIL-valon ja "Hätä Seis"-valon luominen tapahtui vastaavalla tavalla kuin yhtäjaksoisesti palava MIL-valo. MIL\_valo-blokin jälkeen luotiin haara "Hätä Seis"-valolle, jolle määritettiin osoitteeksi 5386.04. Valosta haluttiin tehdä vilkkuva aina, kun kriittinen vika ilmenee, joten "Hätä Seis" (Punainen\_Vika)-blokin jälkeen laitettiin ohjelmassa valmiina ollut yhden sekunnin pulssiblokki (P\_1s). Pulssiblokki muuttaa yhtäjaksoisen signaalin pulssimaiseksi, ja tässä tapauksessa valot vilkkuvat näytöllä sekunnin välein.

Valoille, niin "Hätä Seis"- kuin vilkkuvalle MIL-valolle, täytyi luoda uudet silmukat uusissa osoitteissa. "Hätä Seis"-valolle valikoitui osoitteeksi 25.08, ja MIL-valolle 25.10. Vilkkuva MIL-valo tarvitsi uuden silmukan ja osoitteen, koska se ei olisi toiminut halutulla tavalla "Hätä Seis"-valon kanssa. Kun blokit yhdistettiin toisiinsa, tuloksena oli kuvan 25 mukainen rung.



Kuva 25. MIL-valolle luotu rung kokonaisuudessaan

Kun MIL-valon ja "Hätä Seis"-valon sytyttävät bitit tulevat vikakoodin mukana, rung näyttää kuvan 26 mukaiselta.

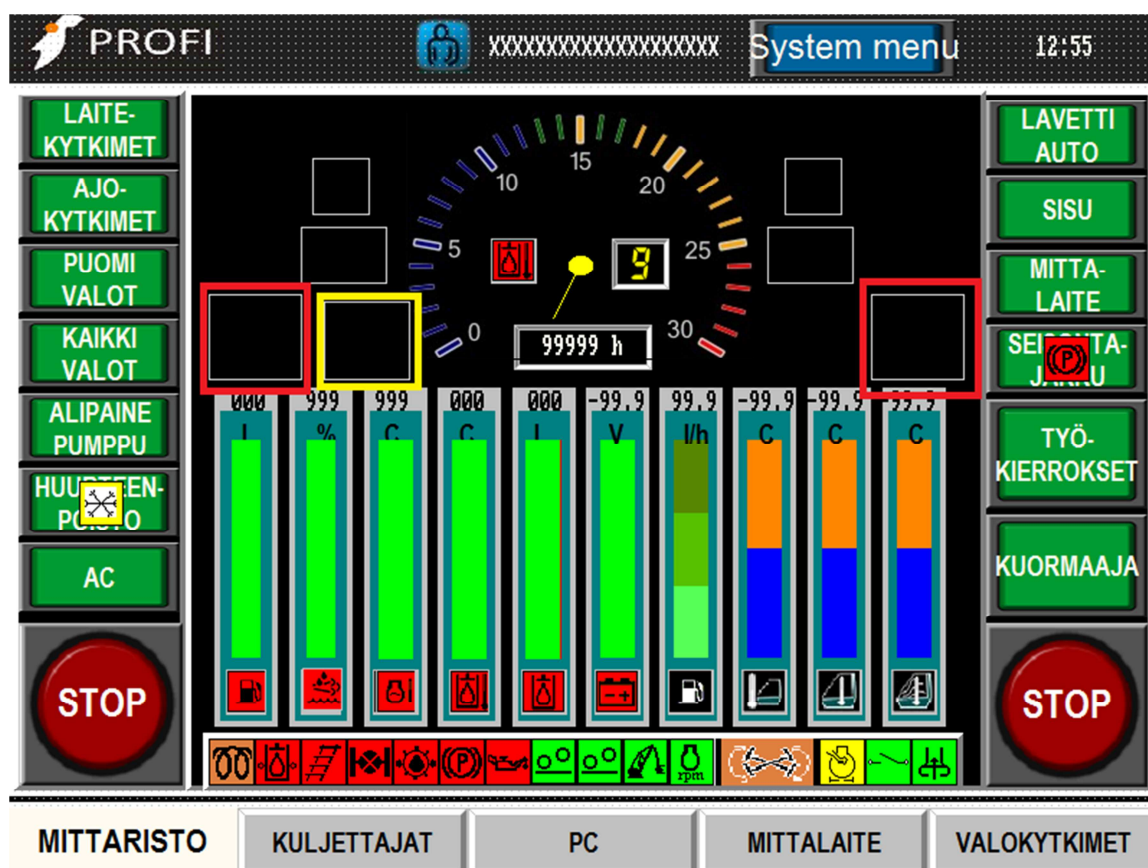


Kuva 26. "Hätä Seis"- ja MIL-valot aktiivisena

Kuten kuvasta 26 voidaan päätellä, vikaindikoinnin syttyminen edellyttää sen, että MIL-valon sytyttävä bitti tulee mukana viestissä. Se, että onko valo yhtäjaksoisesti palava vai vilkkuva, riippuu siitä, tuleeko viestin mukana varoitusvalon sytyttävä bitti vai "Hätä Seis"-valon sytyttävä bitti.

### 6.3 CX-Programmerilla luodun indikoinnin siirtäminen näytölle CX-Designerilla

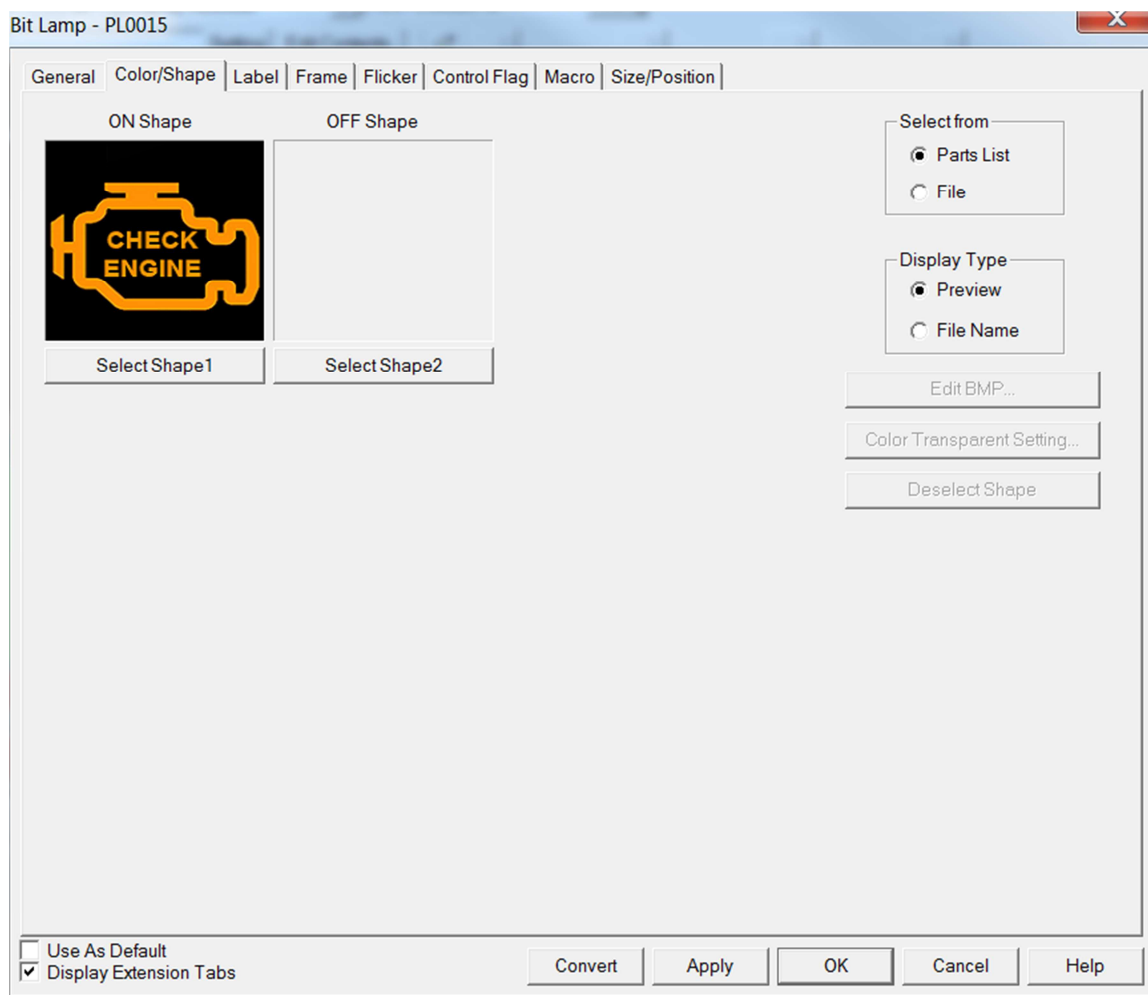
Kun CX-Programmerilla luotu valon indikointi saatiin valmiiksi, pystyttiin tekemään valon indikointi NS-CAA02-näytölle CX-Designerin avulla. Tikapuuverkkoon luodut valot haluttiin saada näkymään näytön alkutilassa eli mittaristossa (kuva x), joten mittariston näkymälle lisättiin kolme uutta bittivaloa (bit lamp). Bittivalot aktivoituvat, kun valon asetuksissa määritetty tietty bitti nousee 0-tilasta 1-tilaan. Vastaavasti bittivalo sammuu, 1-tila muuttuu 0-tilaksi. Nämä bittivalojen sijainnit on korostettu kuvaan 27 väreillä: keltainen on MIL-valoa varten, ja punaiset kriittistä vikaa indikoivaa valoa varten.



Kuva 27. Profi 50 -harvesterin mittaristo korostuksin

### 6.3.1 MIL-valo (varoitus)

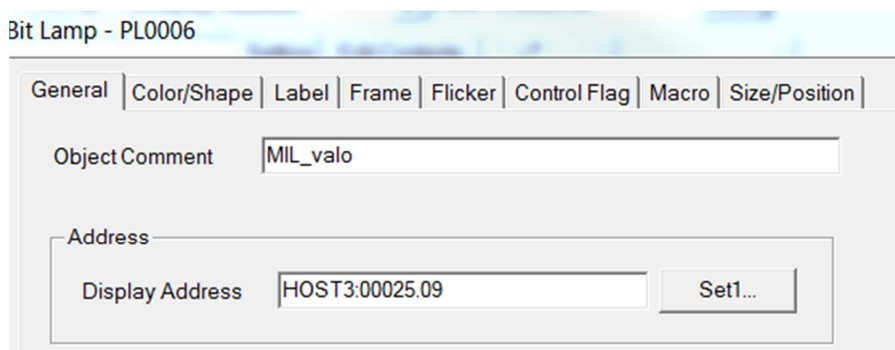
Kun kuvaan 27 laitettut valojen sijainnit saatiin määritettyä, oli seuraava askel valojen kuvan ja osoitepaikan asettaminen. Ensimmäisenä luotiin MIL-valo, joka syttyy kun koneesta tulee viasta varoittava (keltainen valo) viesti. Tällöin MIL-valo palaa yhtäjaksoisesti niin kauan, kunnes vikaviesti häviää moottorilta, aivan kuten tikapuu-verkon poikkipuussa määritettiin. Ensimmäisen muokattiin valon ulkomuoto kuvassa 28 nähtävällä tavalla. Bittivalon asetuksiin pääsee kaksoisnapsauttamalla näytölle luotua bittivaloa, ja ulkomuotoa pystytään muokkaamaan avaamalla Color/Shape-ikkuna.



Kuva 28. MIL-valon muodon määrittäminen

Kuvassa 28 nähdään valon ulkomuodon määrittäminen. Kun vikakoodia ei tule moottorilta, valo on OFF-tilassa (OFF shape). Tällöin mittaristossa ei haluta näkyvän minkäänlaista kuvaa, joten OFF-tilalta poistettiin valon muoto kokonaan.

Vastaavasti vikakoodin tullessa valo syttyy eli menee ON-tilaan (ON shape). Valon muodoksi ladattiin kuva, jota käytetään hyvin yleisesti moderneissa ajoneuvoissa indikoimaan moottorilta tulevaa vikaa. Kun bittivalolle saatiin haluttu ulkomuoto, voitiin valolle määrittää valon sytyttävä bittiosoite kuvan 29 mukaisesti. Osoitetta pystytään muokkaamaan valitsemalla General-ikkuna.

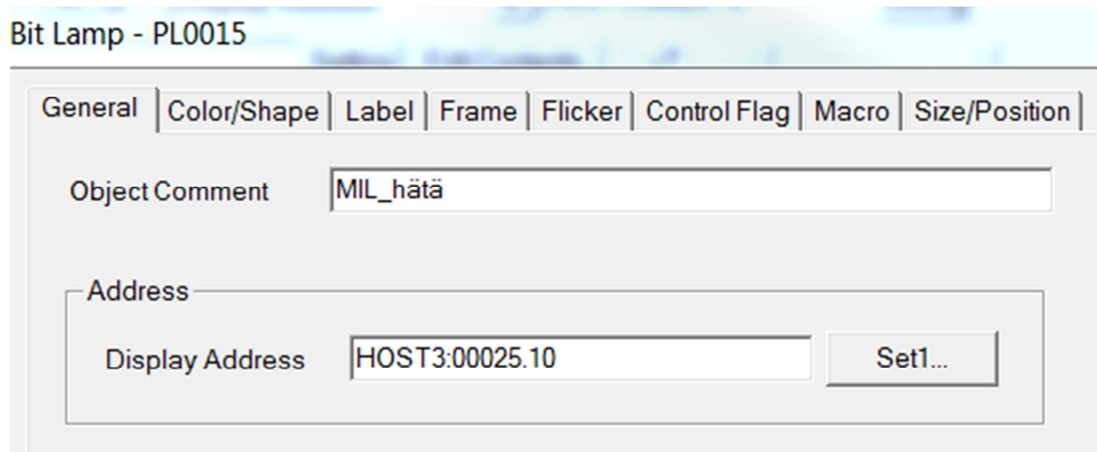


Kuva 29. MIL-valon osoitteen määrittäminen

Kuvassa 29 nähdään MIL-valon osoite (display address) sekä kommentti, jolla voidaan antaa valolle nimi. Osoite-kenttään määritetään laite, josta osoitetta haetaan (tässä tapauksessa HOST3), ja numerot viittaavat osoitteeseen. CX-Programmerilla valon osoitteeksi määritettiin 25.09, joten sama osoite laitetaan tähän kenttään.

### 6.3.2 Vilkkuvat "Hätä Seis"- ja MIL-valot

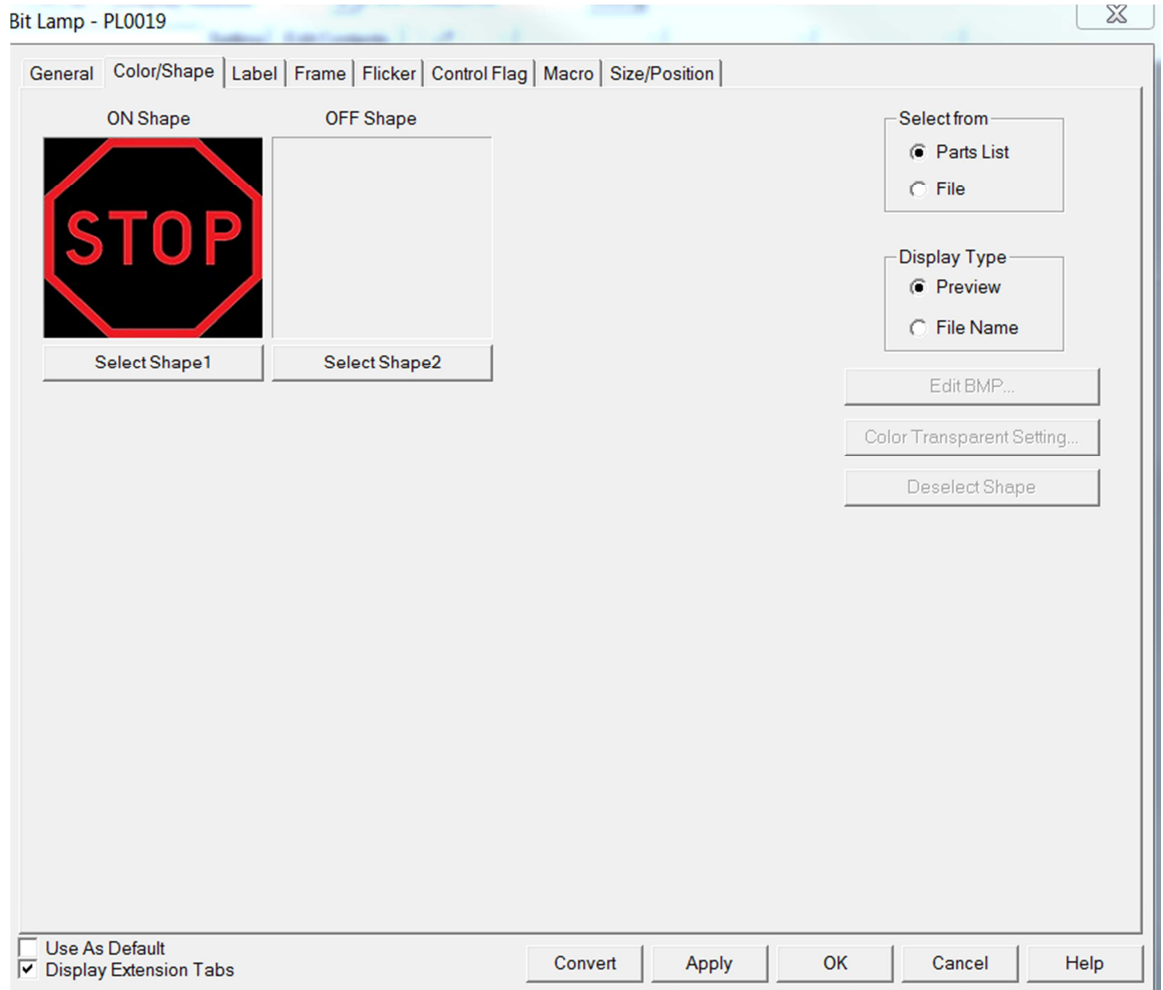
Vilkkuvien "Hätä Seis"- ja MIL-valojen luonti tapahtui vastaavalla tavalla kuin varoittavan MIL-valonkin. Vilkkuva MIL-valo on itse asiassa kopio aiemmin luodusta MIL-valosta, mutta erotuksena on se, että sen syttymistä ja sammumista ohjataan eri osoitteesta, kuten CX-Programmerilla luodussa poikkipuussa määritettiin. Tällöin osoitekenttään tulee kuvan 30 mukainen luku.



Kuva 30. Vilkkuvan MIL-valon osoite

Kuten kuvasta 30 nähdään, ainoat erot kuvassa 29 määritettyihin parametreihin on eri nimi (MIL\_hätä) sekä osoite, joka on 25.10. Vasta luotu MIL\_hätä-valo asetettiin tismalleen samaan kohtaan näytölle kuin varoitusta indikoiva MIL-valo, jolloin MIL-sytyvät tismalleen samassa kohdassa.

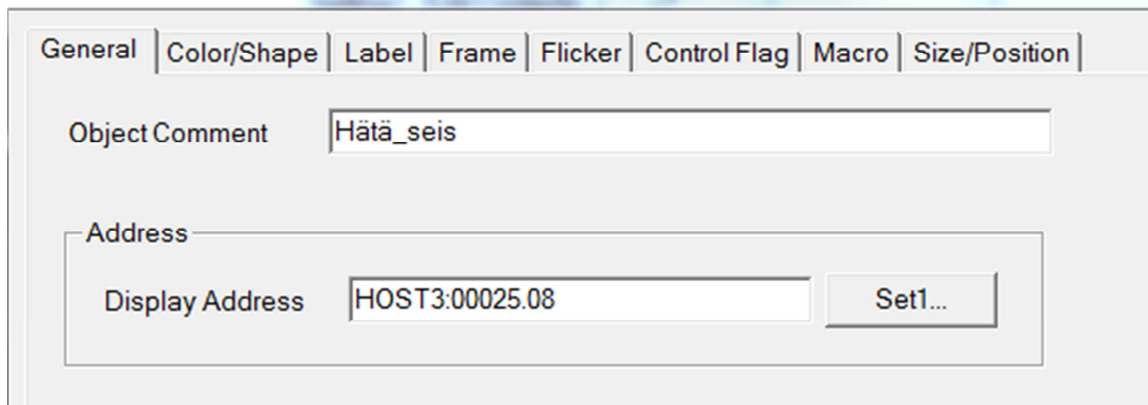
Seuraavaksi luotiin "Hätä Seis"-valo. Valosta haluttiin sellainen, että se indikoi kuljettajalle kyseessä olevan isompi vika sekä sen, että moottori tulee sammuttaa välittömästi. Valon luominen aloitettiin samalla tavalla kuin MIL-valon luominen, eli kaksoisnapsauttamalla bittivaloa päästiin valon asetuksiin käsiksi kuvan 31 osoittamalla tavalla.



Kuva 31. "Hätä Seis"-valon luominen

Kuten MIL-valon kanssa, kuvan OFF-tila laitettiin tyhjäksi. ON-tilan kuvaksi valittiin perinteinen STOP-kuva, jota muokattiin kuvankäsittelyohjelmalla siten, että merkin kehys ja teksti muutettiin punaiseksi ja tausta mustaksi. Tähän päädyttiin siitä syystä, että valo säilyy yhdenmukaisena väriteemaltaan MIL-valon ja muun mittariston kanssa. Kuten MIL-valon kanssa, seuraavaksi määritettiin "Hätä Seis"-valon osoite (kuva 32).

## Bit Lamp - PL0019



General | Color/Shape | Label | Frame | Flicker | Control Flag | Macro | Size/Position

Object Comment: Hätä\_seis

Address

Display Address: HOST3:00025.08 Set1...

Kuva 32. "Hätä Seis"-valon osoite

Valon nimeksi annettiin Hätä-seis, ja osoitteeksi määritettiin 25.08. Kun valo saatiin valmiiksi ja ohjelmaa testattiin näytöllä, yksi "Hätä Seis"-valo tuntui hieman tehottomalta. Tämän vuoksi valosta luotiin identtinen kopio ja se asetettiin toiselle puolelle mittaristoa. Kun moottorilta tulee kriittistä vikaa kuvaava viesti, mittaristo näyttää kuvan 33 mukaiselta.



Kuva 33. Aktiiviset MIL- ja "Hätä Seis"-valot



#### 6.4 Vikakoodien kääntäminen englannista suomeksi

EEM3-spesifikaation mukaiset vikakoodit oli pinottu erilliseen vikalistaan (.text – tiedosto), ja ne löytyivät CX-Designer-ohjelmasta. Vikakoodit olivat sekalaisessa järjestyksessä, joten listaa parannettiin listaamalla viat aihealueittain (moottori, imusarja, virtalähde ja niin edelleen). Näihin vikakoodeihin tehtiin EEM4-spesifikaation edellyttämät muutokset, ja lisäksi ne käännettiin suomen kielelle. Tällöin kuljettajan ollessa suomalainen (jos hän käyttää harvesterin näyttöä suomeksi) hän näkee viat suomeksi, ja vastaavasti englantia käyttävä englanniksi. Muun kielisille käyttäjille (viro, ruotsi, saksa) laitettiin myös englanninkielinen vikalista.

Listasta täytyi saada selville vikakoodeihin liittyvät oleelliset tiedot. Nämä olivat SPN-luku (vikaryhmän numero), FMI-luku (vikatilan tunniste) sekä vian sanallinen selitys. SPN-luku rajaa vika-alueen ja FMI-luku tarkoittaa vian, eli esimerkiksi SPN voi kertoa anturissa olevasta viasta ja FMI tarkoittaa mistä viasta on kyse.

Vikakoodit täytyi kirjoittaa uudelleen Notepad-ohjelmalla, ja tallentaa lista-tiedostona (\*.lst). CX-Designerin oma listausikkuna ymmärtää ainoastaan lst-muotoisen tekstitiedoston, ja PDF-tiedoston kirjoittaminen ja kääntäminen lst-tiedostoon oli hyvin työlästä, loppujen lopuksi tämän insinööriyön työläin osa.

## 7 TESTAUS JA POHDINTA

Työtä testattiin reaaliaikaisesti laboratorio-oloissa, sillä ProfiPro tarjosi tarvittavan laitteiston ohjelmistojen testaamiseen. Ohjelmissa tehdyt muutokset ladattiin PLC:lle, ja vaikutukset pystyttiin näkemään heti näytöltä käsin. PCAN-USB-adapterin avulla moottorilta tulevien vikakoodien simulointi onnistui, ja mittaristolla syttyi biteistä riippuen pelkkä MIL-valo tai vilkkuvat MIL- ja ”Hätä Seis”-valot.

Vikakoodilista toimi sinällään hyvin, mutta ensimmäiset listat näyttivät logiikan näytöllä hyvin sotkuisilta johtuen siitä, että CX-Designerin listausikkuna ei ymmärtänyt tekstitiedostossa tabulaattori-näppäimellä tehtyjä sisennyksiä. Kun sisennykset tehtiin välilyöntiä painamalla, listasta tuli ymmärrettävä. Tämä toimi hyvin niin laboratoriossa kuin varsinaisessa harvesterissakin.

Valojen indikointi ei toiminut harvesterissa odotetulla tavalla. Harvesterin moottorista ei saatu ulos vikaa, joka olisi sytyttänyt sekä MIL- että ”Hätä Seis”-valon bitit. Tästä johtuen ”Hätä Seis”-valo poistettiin kokonaan. MIL-valoa muutettiin siten, että se syttyy aina palamaan, kun keltainen varoitusvalo syttyy. Tämä valo syttyi näytön vikalistassa aina, joten alunäytöllä (mittaristossa) syttyvä MIL-valo indikoi nyt sitä, että koneeseen on tullut uusi vika – eikä vika ole välttämättä moottorissa.

Vikavalon indikointia voitaisiin siis parantaa siten, että valot syttyisivät siten, kuten tässä työssä ne oli alun perin tarkoitus sytyttää. Syy on todennäköisesti moottorilta tulevissa vikakoodeissa ja siinä, että valoja ohjaavat bitit eivät mene 1-tilaan. Luultavasti Agco Powerin henkilöstön avulla valojen indikointi saataisiin toimimaan halutulla tavalla.

Vikakoodeja pystyy parantelemaan siten, että koodit olisivat myös viroksi, saksaksi ja ruotsiksi. Myös näytön käyttäminen venäjäksi olisi etu markkinoilla, mutta nämä muutokset edellyttävät vähintään välttävää kielitaitoa kyseisistä kielistä.

## 8 YHTEENVETO

Insinöörityö onnistui kutakuinkin tavoitteiden osalta, ja pysyi myös aikataulussaan. Vikakoodien indikointia saatiin hieman parannettua kuljettajalle, ja lisäksi vikakoodit näkyvät nyt myös suomeksi.

Työn valmistuminen sujui suhteellisen jouhevissa merkeissä, vaikka sekä automaatio-ohjelmointi että ohjelmoitava logiikkakontrolleri olivat itselleni täysin uusia tuttavuuksia. Myös CAN-väylään täytyi tutustua syvällisemmin, koska CAN-väylä on työssä hyvin olennaisessa osassa. Uusien asioiden opettelu sekä tiedon hankinta otti oman aikansa, mutta loppujen lopuksi suurempia haasteita ei työtä tehdessä ilmennyt.

Kun sain kuulla, että ProfiPro etsii opinnäytetyön tekijää ja työn tekeminen edellyttäisi vikakoodien parissa työskentelemistä, halusin tarttua välittömästi tilaisuuteen. Tulevaisuudessa toivon työskenteleväni insinööritehtävissä, joihin liittyy olennaisesti ajoneuvojen parissa työskentely. Tämä insinöörityö paransi tietojani CAN-väylän toimintaan liittyen ja myös monipuolisti ohjelmointitaitojani. Olen tyytyväinen työn tulokseen, ja uskon tästä insinöörityöstä olevan hyötyä itselleni tulevaisuuden työelämässä.

## LÄHTEET

1. Profi 50-harvesteri [WWW-dokumentti].  
<<http://www.profiopro.fi/index.php/fi/profi-50>>. (Luettu 15.1.2015)
2. ProfiPro Oy [WWW-dokumentti].  
<<http://www.profiopro.fi/index.php/fi/yritys>>. (Luettu 15.1.2015)
3. AGCO Power [WWW-dokumentti].  
<<http://www.agcopower.com/suomi/yritys/>>. (Luettu 15.1.2015)
4. Sisu Valtrassa [WWW-dokumentti]. <<http://www.valtra.com/valtra-engines.aspx>>. (Luettu 22.1.2015)
5. Sisu Massey Ferguson [WWW-dokumentti].  
<[http://www.masseyferguson.com/EMEA/ZA/static/documents/AGCO\\_SISU\\_POWER\\_EN.pdf](http://www.masseyferguson.com/EMEA/ZA/static/documents/AGCO_SISU_POWER_EN.pdf)>. (Luettu 22.1.2015)
6. What is CAN-bus? [WWW-dokumentti]. <<http://canbuskit.com/what.php>>. (Luettu 2.2.2015)
7. What is CAN [WWW-dokumentti]. <[http://www.bosch-semiconductors.de/en/ubk\\_semiconductors/safe/ip\\_modules/what\\_is\\_can/what\\_is\\_can.html](http://www.bosch-semiconductors.de/en/ubk_semiconductors/safe/ip_modules/what_is_can/what_is_can.html)>. (Luettu 15.1.2015)
8. Texas Instrument - Introducing CAN [WWW-dokumentti].  
<<http://www.ti.com/lit/an/sloa101a/sloa101a.pdf>>. (Luettu 8.2.2015)
9. What is SAE J1939 [WWW-dokumentti].  
<<http://www.axiomatic.com/whatissaej1939.pdf>>. (Luettu 15.1.2015)
10. PCAN-USB [WWW-dokumentti]. <<http://www.peak-system.com/PCAN-USB.199.0.html?&L=1>>. (Luettu 15.1.2015)
11. Omronin historia [WWW-dokumentti].  
<<http://www.omron.com/about/corporate/outline/>>. (Luettu 13.2.2015)

12. Omron yrityksenä [WWW-dokumentti].  
<<http://www.omron.com/about/corporate/business/>>. (Luettu 13.2.2015)
13. Omron CJ2M-CPU33 manual [WWW-dokumentti].  
<<http://www.etitudela.com/entrenadorcomunicaciones/downloads/introduccionmanualdeprogramacioncj2.pdf>>. (Luettu 22.2.2015)
14. Automaatio-ohjelmisto CX-Programmer [WWW-dokumentti].  
<[http://industrial.omron.fi/fi/products/catalogue/automation\\_systems/software/configuration/cx-one/cx-programmer.html](http://industrial.omron.fi/fi/products/catalogue/automation_systems/software/configuration/cx-one/cx-programmer.html)>. (Luettu 22.2.2015)
15. CJ1W-CORT21 operational manual [WWW-dokumentti].  
<<http://www.omronkft.hu/nostree/pdfs/plc/networks/w03e-en-02+cj1w-cort21+opermanual.pdf>> (Luettu 22.2.2015)
16. NS operator interfaces [WWW-dokumentti].  
<[http://media.digikey.com/pdf/Data%20Sheets/Omron%20PDFs/NS\\_Operator\\_Interfaces.pdf](http://media.digikey.com/pdf/Data%20Sheets/Omron%20PDFs/NS_Operator_Interfaces.pdf)> (Luettu 22.2.2015)
17. OBD2-Female-connector [WWW-dokumentti].  
<<http://shop.dieselschrauber.de/obd2-female-connector-can-kit-p-318-l-en.php>>. (Luettu 4.3.2015)